

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

Не все сервисы имеет смысл постоянно выставлять в мир. Но, однако, доступ из мира к ним все же иногда нужно иметь. Можно, конечно, зайти на сервер, например, по VPN и временно изменить конфигурацию, сделать необходимые действия и вернуть конфигурацию назад.

Для упрощения таких действий есть довольно простая, но, тем не менее, богатая возможностями, технология Port Knocking. Ее суть заключается в том, что послав пакеты в определенной последовательности на определенные порты, вы инициируете какое-то наперед сконфигурированное действие. Настройку Port Knocking можно сделать самому в полностью ручном режиме при помощи IPTables, однако проще и удобнее для этого использовать специальный демон knockd.

Установим его командой:

```
# apt-get install knockd
```

□

Настройки сервиса делаются в файле `/etc/knockd.conf`

По умолчанию он выглядит так:

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

*[options]*

*UseSyslog*

*[openSSH]*

*sequence = 7000,8000,9000*

*seq\_timeout = 15*

*command = /sbin/iptables -A INPUT -s %IP% -p tcp --dport 22 -j ACCEPT*

*tcpflags = syn*

*[closeSSH]*

*sequence = 9000,8000,7000*

*seq\_timeout = 5*

*command = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT*

*tcpflags = syn*

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

□

В этом файле описаны два действия — открыть порт для ssh и закрыть его же.

Описание полей:

*sequence* — список портов, на которые нужно «постучать»

*seq\_timeout* — максимальное время в секундах, которое отводится на «простукивание» портов

*command*— команда, которая будет выполнена

*tcpflags* — флаги пакетов

После установки так же следует разрешить запуск демона knockd. Для этого в файле */etc*

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

*c/default/knockd*

прописать строку:

```
START_KNOCKD=1
```

□

Однако особого смысла в этих настройках пока что нет. В IPTables по умолчанию все порты открыты. Для того, что бы не отслеживать последовательность правил фильтрации, в таблице FILTER в цепочке INPUT зададим по умолчанию запрещающее правило:

```
# iptables -P INPUT DROP
```

□

И перезапустим сервис:

```
#service knockd restart
```

□

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

Теперь у нас по умолчанию все порты закрыты. Для того, что бы соединиться с сервером нужно к нему постучаться. Конечно, это можно сделать и при помощи telnet, но есть более удобный путь.

Если мы пытаемся соединиться с Linux-компьютера, то установим на нем тот же самый knockd, но не будем запускать сервис, а воспользуемся клиентской частью:

```
$ knock 192.168.0.140 7000 8000 9000
```

□

После чего можно соединяться с нашим сервером:

```
$ ssh user@ 192.168.7.140
```

□

Если мы используем Windows-компьютер, то можно скачать knock-утилиту

[www.zeroflux.org/proj/knock/files/knock-cygwin.zip](http://www.zeroflux.org/proj/knock/files/knock-cygwin.zip)

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

□

И после ее запуска из командной строки (параметры такие же как и в Linux) соединиться с сервером при помощи Putty как обычно.

После того, как мы «постучались» в syslog должны появиться записи типа:

```
knockd: 192.168.0.5: openSSH: Stage 1
```

```
knockd: 192.168.0.5: openSSH: Stage 2
```

```
knockd: 192.168.0.5: openSSH: Stage 3
```

```
knockd: 192.168.0.5: openSSH: OPEN SESAME
```

```
knockd: openSSH: running command: /sbin/iptables -A INPUT -s 192.168.0.5 -p tcp --dport 22 -j ACCEPT
```

□

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

После окончания работы нужно не забыть закрыть порт:

```
$ knock 192.168.0.140 9000 8000 7000
```

□

После этого knockd выполнит удаление правила на открытие 22 порта.

У такого подхода есть некоторые недостатки — нельзя просто оборвать сессию с сервером и продолжить заниматься другими делами. В таком случае порт на сервере будет открыт или до перезагрузки, или пока мы не вспомним и не отменим разрешающее правило. Что бы избежать такой ситуации правило на «стук» по портам можно записать в несколько другом виде.

Приведем файл `/etc/knockd.conf` к следующему виду:

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

*[options]*

*UseSyslog*

*[openSSH]*

*sequence = 7000,8000,9000*

*seq\_timeout = 15*

*tcpflags = syn*

*start\_command = /sbin/iptables -A INPUT -s %IP% -p tcp --dport 22 -j ACCEPT*

*cmd\_timeout = 10*

*stop\_command = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT*

□



## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

Сейчас у нас несколько изменилось описание реакции на «стук» по портам:

*start\_command* — команда, которая будет выполнена

*cmd\_timeout* — время задержки в секундах

*stop\_command* — команда, которая будет выполнена через *cmd\_timeout* секунд после выполнения *start\_command*.

□

И перезапустим сервис knockd

```
#service knockd restart
```

□

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

Если мы сейчас попробуем «постучаться» и присоединиться к серверу, то через 10 секунд наша сессия оборвется, так как сработает правило закрытия порта, и не факт, что мы успеем сделать все, что нужно.

Для того, что бы сессия не обрывалась на сервере в IPTables нужно добавить правило, которое будет разрешать входящий трафик, при условии, что он относится к уже установленному соединению:

```
# iptables -A INPUT -i eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT
```

□

Вот после этого мы сможем «постучаться» и после этого нормально открыть сессию и поработать сколько нужно. Однако через 10 секунд новую сессию открыть мы уже не сможем.

После того, как мы все настроили можно спокойно посмотреть [заготовки на зиму рецепты с фото](#)

## Port Knocking

Автор: yakim

24.07.2013 18:20 - Обновлено 24.07.2013 18:27

---

{comments on}